AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method of designing an essentially digital system, the method comprising: generating a system-level description of the functionality and timing of ~~a~~ the digital system, the system-level description comprising a plurality of tasks; optimizing task concurrency in the system-level description to obtain a task concurrency optimized system-level description that includes ~~at least~~ partly Pareto task optimization information; and designing the essentially digital system based at least in part upon the task concurrency optimized system-level description, wherein the task concurrency optimized system-level description further includes a description of a real-time operating system that uses the Pareto task optimization information.

2. Cancelled.

3. (Previously Presented) A method of designing an essentially digital system, the method comprising:

generating a description of the functionality and timing of the digital system, wherein the description includes a grey-box system-level description comprising a plurality of tasks, and wherein the grey-box system level description comprises a multi-thread graph for inter-task descriptions and a control data flow graph for intra-task descriptions;

optimizing task concurrency in the grey-box system-level description, thereby obtaining a task concurrency optimized grey-box system-level description; and

designing the essentially digital system based at least in part upon the task concurrency optimized grey-box system-level description.

4. (Original) The method of claim 3, wherein the task concurrency optimized system-level description further includes a description of a real-time operating system.

5. (Previously Presented) The method of claim 1, 3 or 4 wherein optimizing task concurrency comprises separately performing design-time intra-task scheduling for at least two of the tasks, thereby generating a plurality of intra-task schedules for each of the tasks.

6. (Original) The method of claim 5, wherein the plurality of intra-task schedules are subset of all possible intra-task schedules, wherein the subset includes Pareto optimal schedules.

7. (Previously Presented) The method of claim 1 or 4 wherein optimizing task concurrency comprises designing a run-time scheduler that is part of the real-time operating

system, wherein the run-time scheduler dynamically schedules at least two of the plurality of tasks.

8. (Cancelled).

9. (Original) The method of claim 5, wherein the digital system comprises a plurality of processors, and wherein the design-time intra-task scheduling uses processor power consumption optimization information to assign at least one of the tasks to at least one of the processors.

10. (Previously Presented) The method of claim 9, wherein at least one processor is a multi-supply voltage processor.

11. (Currently Amended) A method of designing an essentially digital system, the method comprising:

generating a system-level description of the functionality and timing of the digital system, the system-level description comprising a plurality of tasks;

optimizing task concurrency in the system-level description by separately performing design-time intra-task scheduling for at least two of the tasks~~,~~ to generate a plurality of intra-task schedules for each of the tasks, wherein the plurality of intra-task schedules is a subset of all possible intra-task schedules, the subset including ~~at least~~ partly Pareto optimal schedules;

obtaining a task concurrency optimized system-level description, including at least partly Pareto task optimization information, the subset defining the Pareto task optimization information; and

designing the essentially digital system based on the task concurrency optimized system-level description.

12. (Currently Amended)A method of designing an essentially digital system, the method comprising: generating a system-level description of the functionality and timing of a digital system, the system-level description comprising a plurality of tasks; optimizing task concurrency in the system-level description to obtain a task concurrency optimized system-level description that includes ~~at least~~ partly Pareto task optimization information; and designing the essentially digital system based at least in part upon the task concurrency optimized system-level description,

wherein non-deterministic behavior of the digital system is modeled by interacting the tasks, ~~while~~ wherein each of the tasks describe part of the deterministic behavior of the digital system, and

wherein optimizing task concurrency comprises designing a run-time scheduler that is part of the real-time operating system, wherein the real-time operating system is capable of performing dynamic scheduling of at least two of the tasks.

13. (Currently Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform the method, comprising: optimizing task concurrency in a system-level description of the functionality and timing of a digital system, wherein the system-level description comprises a plurality of tasks, wherein optimizing includes separately performing design-time intra-task scheduling for at least two of the tasks to generate a plurality of intra-task schedules for each of the tasks, wherein the plurality of intra-task schedules are a subset of all possible intra-task schedules, and wherein the subset defines ~~at least~~ partly Pareto task optimization information.

14. (Currently Amended) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform the method comprising: selecting one or more schedules for a plurality of tasks from a plurality of ~~at least~~ partly Pareto optimal intra-task schedules; and executing one of the tasks in accordance with the selected schedule.

15. (Previously Presented) The method of claim 5, wherein the digital system comprises at least one processor, and wherein the design-time intra-task scheduling uses processor power consumption optimization information to determine an appropriate scheduling.

16. (Previously Presented) A method of designing an essentially digital system, the method comprising: generating a system-level description of the functionality and timing of the digital system, the system-level description comprising a plurality of tasks; optimizing task concurrency in the system-level description, to obtain a task concurrency optimized system-level description that includes cost-cycle budget tradeoff information; and designing the essentially digital system based at least in part upon the task concurrency optimized system-level description, wherein the task concurrency optimized system-level description further includes a description of a real-time operating system that uses the cost-cycle budget tradeoff information.

17. (Previously Presented) A method of designing an essentially digital system, the method comprising:

generating a system-level description of the functionality and timing of the digital system, the system-level description comprising a plurality of tasks;

optimizing task concurrency in the system-level description by separately performing design-time intra-task scheduling for at least two of the tasks, to generate a plurality of intra-task schedules for each of the tasks, wherein the plurality of intra-task schedules is a subset of all possible intra-task schedules, the subset including cost-cycle budget tradeoff information;

obtaining a task concurrency optimized system-level description, including cost-cycle budget tradeoff information; and

designing the essentially digital system based on the task concurrency optimized system-level description.

18. (Previously Presented) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method, comprising: optimizing task concurrency in a system-level description of the functionality and timing of a digital system, wherein the system-level description comprises a plurality of tasks, wherein optimizing includes separately performing design-time intra-task scheduling for at least two of the tasks to generate a plurality of intra-task schedules for each of the tasks, wherein the plurality of intra-task schedules are a subset of all possible intra-task schedules, and wherein the subset defines cost-cycle budget tradeoff information.

19. (Previously Presented) A program storage device readable by a machine, tangibly embodying a program of instructions executable by the machine to perform a method comprising: selecting one or more schedules for a plurality of tasks from a plurality of intra-task schedules based upon cost-cycle budget tradeoff information; and executing one of the tasks in accordance with the selected schedule.

20. (Previously Presented) A method of designing an essentially digital system, the method comprising:

generating a description of the functionality and timing of the digital system, wherein the description includes a grey-box system-level description comprising a plurality of tasks;

optimizing task concurrency in the grey-box system-level description to obtain a task concurrency optimized grey-box system-level description; and

designing the essentially digital system based at least in part upon the task concurrency optimized grey-box system-level description,

wherein non-deterministic behavior of the digital system is modeled by interacting the tasks, while each of the tasks describe part of the deterministic behavior of the digital system, and

wherein optimizing task concurrency comprises designing a run-time scheduler that is part of the real-time operating system, wherein the real-time operating system is capable of performing dynamic scheduling of at least two of the tasks.